REMARKS/ARGUMENTS

Claims 1-20 were pending and variously rejected under 103(a) by Gossweiler ('327) and Gossweiler in view of Harvill ('845). New claims 21-23 are added.

Claim 6 was objected to. In response, claim 6 has been amended to properly recite that the asset management system receives a specification for an object, and responds with an object location. Specifically, claim 6 now recites:

providing to an asset management system the first specification of the object; and

receiving a location of the first representation of the one object from the asset management system.

Additionally, claim 14 was objected to and was amended to replace a period with a semicolon.

Amendments have also been made to claims 4,5,12,13, and 18 for stylistic purposes to place the claims in a better-recognized Markush claim format. No new matter has been added thereby

I. THE PRESENT INVENTION

The present invention relates to improved methods and apparatus for efficiently processing complex scenes.

As discussed in the background of the present invention, in the Industry, a single scene descriptor (file) was typically used to describe a scene, and typically included data describing all objects, all camera angles, all lighting sources, etc. within the scene. [0008], l.1-3. After an entire scene descriptor was loaded into memory, the data specified in the scene descriptor was used to render the scene. [0008], l.3.

As it is widely recognized in the Industry, Pixar has been on the forefront of computer animation. By doing so, Pixar and been pushing the limits of technology, by specifying more complex scenes, using more lights, specifying more cameras, and including more objects. In light of their experiences with such cutting-edge complexity, Pixar discovered that using a single "monolithic" scene descriptor to store all scene data, had numerous disadvantageous. As stated in the specification:

[0009] Drawbacks with the above techniques include that there are a great number of objects in a typical scene, each typically having a great number of parameters that can or must be set by the animator. The scene data file (also known as a scene descriptor file) that describes the entire scene is accordingly, typically very large

(gigabytes). Because of the sizes of typical scene descriptor files are typically large, the animator's computer must have sufficient amount of memory. As an another drawback, because scene descriptor files are typically large, the animators typically have to wait for the descriptor files to be retried from memory. Additionally, because scene descriptor files are typically large, the animators typically have to wait a long time for the scene to be rendered.

In light of these problems, the inventor came up with the idea of breaking-up scene descriptor files. Specifically, the inventor developed the notion of "hook files" and "hook sets" where a scene descriptor file (a hook-set) is stored separately from object data, camera data, lighting data. In such cases, the scene descriptor file merely includes references to file names (hook files) where object data, camera data, and lighting data could be found. [0036], l.4.

As also described, values of properties of the objects, cameras, and lights are specified in the scene descriptor file. For example, [0055]-[0081] illustrate parameters of a camera; [0094]-[0108] illustrate parameters of a light; and [0121] and [0122] specify parameters of an object (e.g. number_of_legs=4, number_of_legs=1). Another example of the scene descriptor including object parameters, but not the entire object data is illustrated in [0137]-[0139]

In these discussed embodiments, the object data, camera data, and the lighting data were specified in terms of a path to a filename.

In using this solution, the inventor recognized that "hard coded" locations (e.g. directory paths) for hook files was technically restrictive. Specifically, the specification states:

[0147] ... Drawbacks include that it requires the user to understand the computer directory structure for objects. Further, it requires the user to know how different representations of objects are classified and stored within the directory structure. Other drawbacks include that as a directory structures changes, the animator who creates the scene descriptor file must constantly modify the scene descriptor file to stay current with such changes. Still other drawbacks include that over time, and with computer system migrations, or the like, the object directory structures may become corrupt, and objects may be lost. As a result, years after a scene is initially rendered, when the user attempts to re-render the scene, not all objects will be found.

In light of these drawbacks, the inventor developed additional embodiments where the assets, e.g. object data, camera data, lighting data, and the like were retrievable from an asset management system and/or a database, as illustrated in Fig. 4:

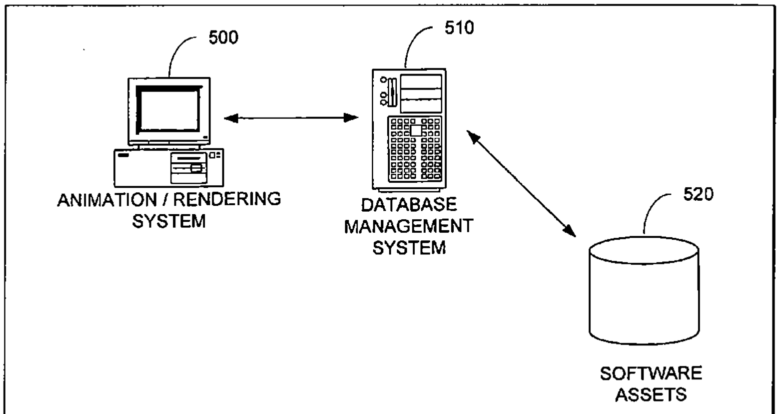


FIG. 4

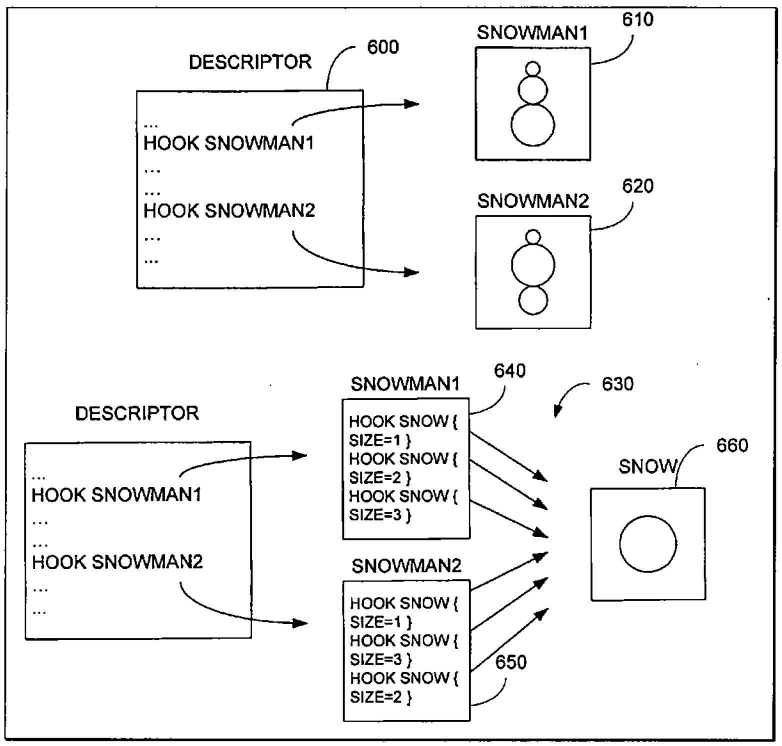
As stated in the specification, the use of an asset management system and / or a database greatly reduces the complications of changes and updates to objects, cameras and lighting data, made by other users. Specifically:

> *[0067] These examples demonstrate that these embodiments should greatly reduce the amount of work required to manage the scene descriptor file when object representations change or are updated, when new objects are added to the scene, or the like. Accordingly, the inventor believes that the advantages of these embodiments greatly enhance the value of embodiments of the present invention.*

Thus, as stated above, the use of an asset management and /or a database for storage for the above-described data provided additional advantages to the inventor. For example, efficient access to the many objects in a scene, management of revisions, management of versions, management of resources to the many different users who define a scene, and the like. In other words, not only were scene descriptors now manageable in terms of computer hardware resources, but also, the component portions of the scene descriptors could be managed in terms of project management.

Aspects of the above embodiment, is believed to be recited in various claims. For example, Claim 1 recites, in part, querying a database for a first representation of the one object in response to the first specification of the object when the selection is of the first rendering option. Claim 1 also recites wherein the first representation of the object is not loaded into computer memory when the selection is of the second rendering option.

According to another embodiment of the invention described in the specification, "nuggets" are discussed in conjunction with Fig. 5. In Fig. 5, a first snowman object 610 and a second snowman object 620 are specified in a scene descriptor hook-set 600. As can be seen the first snowman object 640 may itself be a hook-set that references a sphere object 660 and specifies appropriate parameters (e.g. size=1, size=2, size=3), and the second snowman object 650 may itself be a hook-set that references a sphere object 660 and specifies appropriate parameters (e.g. size=1, size=2, size=2).



As stated in the specification, by having objects being able to decompose to smaller objects, and even the same object, e.g. object 660, the amount of information used to describe a scene may be greatly reduced. [0154], l.5-7. Additionally, the specification states:

> *[0071] In embodiments of the present invention, the nuggets concept also <u>greatly increases the ease and ability for changes to be propagated for objects</u>. For example, as illustrated in FIG. 5, if sphere object 660 is changed to a cube object 670, when retrieved in the scene, snowman object 680 and snowman object 690 are now made-up of cubes not spheres. Thus, as illustrated, a single change to a geometric primitive, is easily propagated through out the scene.*

Aspects of the above embodiment, is believed to be recited in various claims. For example, Claim 8 recites in part, the first representation of the object comprises references to representations of a first plurality of objects, wherein the second representation of the object comprises references to representations of a second plurality of objects, and <u>wherein at least one object within the first plurality of objects is also within the second plurality of objects</u>.

II. THE CITED REFERENCES

A. Gossweiler

Gossweiler discloses a system where different level of details (LOD) for objects in a scene may be selected based upon available processing resources.

A major portion of the problem in Gossweiler is directed to determining the optimal graphics processing fraction, for example steps 502-606 in Figs. 5-6. Gossweiler initially discloses a "quadratic parametric equation" is solved to determine an optimal graphics processing <u>fraction</u> between two objects A and B, Abstract, l. 15-18. Figs. 5-6: Steps 502-606. Based upon the graphics processing fractions, appropriate LOD model for objects A and B are selected. Abstract, l. 18-21. Fig. 7: Step 703, 706.

In the case object A or B include children objects, the appropriate LOD models for the children objects are also selected. Abstract, l. 25-30. Fig. 7: Steps 702, 705. Based upon the selected LOD models, the objects are then rendered. Abstract, l. 30-32. Fig. 7: Step 708. A major portion of the problem in Gossweiler is directed to determining the optimal graphics processing fraction, for example steps 502-606 in Figs. 5-6.

Gossweiler appears to be silent as to how objects are stored with respect to a scene 400. Gossweiler merely discloses that a scene 400 may include multiple objects such as objects 201 and 411, which can be represented by different LOD representations. However, Gossweiler, does not mention how LODs 404-406 are stored, or how LODs 407-410 are stored, for example.

The undersigned believes that Gossweiler appears to simply use a "monolithic" scene descriptor that is disclosed in the background of the present invention. Gossweiler does not mention that a scene descriptor can be separated into separate data files. Specifically, Gossweiler makes no mention about LOD 404, 405, and 406 being separate from a scene descriptor file 400. Gossweiler appears to state the opposite, "The scene object contains all of the leaf objects in the scene." Col. 9. l.34-35.

B. Harvill

Harvill relates to a method for transferring animation data over a "low band-width" communications network. For example, Harvill discloses initially downloading "an initial animation download package" including the object and basic actions, and then streaming additional "behavior data." Col. 2, lines 24-26, l.29-33. Specifically, Harvill discloses:

> *In this manner, the client computer may more quickly begin the animation while other yet unneeded actions or behaviors are being downloaded. col. 2, l.36-38.*

Harvill fails to disclose anything about a scene descriptor, and does not even address the problem of how to store and retrieve massive amounts of scene descriptor.

III. THE CITED REFERENCES DISTINGUISHED

A. Claim 1

Gossweiler fails to disclose all the limitations of claim 1. For example, Gossweiler fails to disclose querying a database for a first representation of the one object in response to the first specification of the object when the selection is of the first rendering option. Additionally Gossweiler fails to disclose wherein the first representation of the object is not loaded into computer memory when the selection is of the second rendering option.

As discussed above, Gossweiler is silent as to how LOD models are stored. It is assumed Gossweiler merely uses a conventional single "monolithic" scene descriptor of the prior art to store a scene, including object models. If so, Gossweiler would not store the different LOD models in a separate location, and thus, no database would be used to store models.

Additionally, if Gossweiler merely uses a conventional "monolithic" scene descriptor of the prior art to store a scene, Gossweiler would load the entire scene, including all the different LOD models into memory at one time. Because Gossweiler would not know which LOD models were required for rendering purposes, except at run-time based upon the resources currently available, all the different LOD models would most likely be available for rendering from memory.

Additionally, the Undersigned traverses the Examiner's statement that it would have been obvious to combine Gossweiler with a database. Although databases were well-known, in the field of computer animation, the use of databases for storing objects described within a scene descriptor, as described in the specification, was not obvious. Instead, it is reiterated that typical scene descriptors at the time of the present invention were very large (gigabytes) because they stored all data within a scene. [0008] l.1-9.

In light of the above, and for other reasons, claim 1 is not obvious in light of Gossweiler.

B. Claim 8

Gossweiler fails to disclose all the limitations of claim 8. For example, Gossweiler fails to disclose wherein the first representation of the object is not loaded into computer memory when the selection is of the second rendering option.

As discussed above, with typical "monolithic" scene descriptors all objects, all of Gossweiler's LOD models would be loaded into memory. In contrast, as recited above, not all objects are loaded into computer memory.

Additionally, Gossweiler also fails to disclose wherein the first representation of the object comprises references to representations of a first plurality of objects, wherein the second representation of the object comprises references to representations of a second plurality of objects, and wherein at least one object within the first plurality of objects is also within the second plurality of objects. Specifically, Gossweiler fails to disclose two separate objects, each including references to the same objects, as recited above.

In light of the above, and for other reasons, claim 8 is not obvious in light of Gossweiler.

C. Remaining Claims

Dependent claims 2-7,21-23 are asserted to be allowable for substantially the same reasons as claim 1, and more particularly, for the specific limitations they recite.

Dependent claims 8-13 are asserted to be allowable for substantially the same reasons as claim 7, and more particularly, for the specific limitations they recite.

Independent claim 14 is asserted to be allowable for substantially the same reasons as claim 7, and more particularly, for the specific limitations it recites.

Dependent claims 15-20 are asserted to be allowable for substantially the same reasons as claim 14, and more particularly, for the specific limitations they recite.

CONCLUSION

In view of the foregoing, Applicants believe all claims now pending in this Application are in condition for allowance. The issuance of a formal Notice of Allowance at an early date is respectfully requested.

If the Examiner believes a telephone conference would expedite prosecution of this application, please telephone the undersigned at (650) 326-2400.

Respectfully submitted,

Stephen Y. Pang
Reg. No. 38,575

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, California 94111-3834
Tel: (650) 326-2400 Fax: (650) 326-2422
SYP:djb
60757188 v1